

Lesson 1: History of the Web

1.1 Evolution of the Web

The World Wide Web (WWW) has undergone significant transformations since its inception. Understanding its history provides insight into how modern web development has emerged.

- **1989–1990: The Birth of the Web**

- Tim Berners-Lee proposed the concept of the World Wide Web while working at CERN.
- Key innovations included HTML (HyperText Markup Language), HTTP (HyperText Transfer Protocol), and URLs (Uniform Resource Locators).

- **1990s: Early Development**

- The first web browser, WorldWideWeb (later renamed Nexus), was created.
- Introduction of static web pages and table-based layouts.
- Web directories like Yahoo! emerged, making information more accessible.

Additional Details:

- Mosaic, one of the first graphical web browsers, was introduced in 1993, which greatly increased public interest in the web.
- By the mid-1990s, e-commerce platforms like Amazon and eBay began to appear, setting the stage for online business.

- **2000s: Web 2.0**

- Focus shifted to interactivity, user-generated content, and social networking.
- Technologies like JavaScript, CSS, and AJAX became prominent, enabling dynamic websites.
- Blogging platforms like WordPress and social media sites like Facebook transformed content sharing.

Additional Details:

- The concept of "Web 2.0" emphasized collaboration, with the rise of wikis, forums, and video sharing platforms like YouTube.
- Rich Internet Applications (RIAs) became popular, leveraging Flash and Silverlight before HTML5 took prominence.

- **2010s: Mobile-First and Modern Web**

- Rise of mobile-friendly websites and responsive design to cater to smartphone users.
- Introduction of frameworks (e.g., React, Angular, Vue.js) and APIs for streamlined development.
- Cloud computing gained traction, facilitating scalable web applications.

Additional Details:

- Mobile apps often worked alongside web platforms to create seamless user experiences.
- The push for accessibility standards (e.g., WCAG) ensured more inclusive web experiences.

- **Today: Web 3.0 and Beyond**

- Emphasis on decentralization, blockchain technologies, and artificial intelligence.
- Immersive technologies such as Augmented Reality (AR) and Virtual Reality (VR) are integrated into web experiences.
- The use of Progressive Web Apps (PWAs) bridges the gap between web and native applications.

Additional Details:

- Web 3.0 focuses on data ownership and privacy, leveraging decentralized networks.
- AI-driven personalization tailors web content to individual users.

1.2 Key Technologies and Their Development

- **HTML:** Provides the structure of web pages and ensures content is displayed correctly.
 - HTML uses elements like `<head>` for metadata, `<body>` for content, and semantic tags like `<header>` and `<footer>` to improve accessibility.
 - The transition from HTML4 to HTML5 introduced features like multimedia support (`<audio>`, `<video>`) and semantic elements (`<article>`, `<section>`).
- **CSS:** Handles the styling and layout, making websites visually appealing.
 - CSS evolved from inline styles to modern techniques like external stylesheets and pre-processors (e.g., SASS, LESS).
 - CSS3 introduced advanced features like animations, gradients, and media queries for responsive design.
- **JavaScript:** Adds interactivity, enabling features like animations, form validation, and dynamic content updates.
 - JavaScript frameworks (e.g., React, Angular, Vue.js) simplify development and enable Single Page Applications (SPAs).
 - ES6 (ECMAScript 2015) introduced significant updates like arrow functions, modules, and classes.
- **Web Servers:** Deliver content to users by processing requests and sending responses (e.g., Apache, NGINX).
 - Servers use protocols like HTTP/HTTPS to communicate with clients.
 - Modern trends include serverless architectures and containerization (e.g., Docker).
- **Browsers:** Tools for accessing web content, such as Google Chrome, Mozilla Firefox, and Microsoft Edge.
 - Browsers render HTML, CSS, and JavaScript to display web pages.
 - Developer tools in modern browsers aid in debugging and optimizing web applications.

Lesson 2: Internet Fundamentals

2.1 The Internet and How It Works

The Internet is a global network that connects millions of devices, enabling communication and data exchange.

- **Client-Server Model:**

- The client (e.g., a browser) sends a request to a server.
- The server processes the request and sends back the response, such as a web page or file.

Additional Details:

- Clients include devices like laptops, smartphones, or IoT devices.
- Servers can host applications, store data, or provide APIs for third-party use.

- **Key Components:**

- **IP Address:** Unique identifier for devices on a network.
- **DNS (Domain Name System):** Translates human-readable domain names into IP addresses.

2.2 URL Structure

URLs (Uniform Resource Locators) are used to locate resources on the web.

- Components of a URL:

- **Protocol:** Defines how data is transferred (e.g., `http`, `https`).
- **Domain Name:** Identifies the server (e.g., `example.com`).
- **Path:** Specifies the resource location on the server (e.g., `/about`).

Example:

```
https://www.example.com/about
Protocol: https
Domain: www.example.com
Path: /about
```

Additional Details:

- Query strings (e.g., `?id=123`) allow for dynamic data retrieval.
- Fragments (e.g., `#section`) point to specific parts of a web page.

2.3 Anatomy of a Web Page

- **HTML Document:** Forms the basic structure.

- Example:

```
<!DOCTYPE html>
<html>
  <head>
    <title>My First Web Page</title>
  </head>
```

```
<body>
  <h1>Welcome to My Website</h1>
  <p>This is a simple web page.</p>
</body>
</html>
```

- **Styling with CSS:** Adds layout and design.
 - **Interactivity with JavaScript:** Enhances functionality.
-

Lesson 3: HTML Basics

3.1 HTML Document Structure

HTML documents follow a specific structure that ensures proper organization and rendering.

- **Basic Structure:**

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document Title</title>
  </head>
  <body>
    <!-- Content goes here -->
  </body>
</html>
```

3.2 Common HTML Elements

- **Headings (`<h1>`** to `<h6>`)**: Define content hierarchy.
- **Paragraphs (`<p>`):** Group text into blocks.
- **Links (`<a>`):** Create hyperlinks.
 - Example:

```
<a href="https://www.example.com">Visit Example</a>
```

- **Images (``):** Embed visuals.
 - Example:

```

```

3.3 Best Practices

- Use semantic HTML for accessibility and SEO.
 - Always include alternative text for images.
 - Validate your HTML code using tools like W3C Validator.
-

Questions for Students

Lesson 1: History of the Web

1. Who invented the World Wide Web, and what were its key innovations?
2. What are the major differences between Web 1.0, Web 2.0, and Web 3.0?
3. How did the introduction of JavaScript and CSS impact web development?

Lesson 2: Internet Fundamentals

1. Explain the client-server model in your own words.
2. What is the role of DNS in the Internet?
3. Break down the URL <https://www.school.com/courses> and identify its protocol, domain, and path.

Lesson 3: HTML Basics

1. What is the purpose of the `<!DOCTYPE html>` declaration?
2. Write the basic structure of an HTML document from memory.
3. What is the difference between a `<div>` and a ``?
4. Why is it important to include alternative text (`alt`) for images?